

METHOD AND APPARATUS FOR DETERMINING A PORTION OF  
TOTAL COSTS OF AN ENTITY

5

Background of the Invention:

Field of the Invention:

The present invention relates to the field of distribution of costs in a system, in which multiple users or entities have access to a product. In particular, the present invention relates to such a system, in which the total costs are related to a total maximum number of accesses of all entities occurring in a predetermined time window over a time interval, and in which a fair and transparent distribution of the costs among the participating entities is required.

Licenses are an example of a kind of an access to a product that are very expensive. There are two places to influence their costs: (1) the contract that determines the cost model underlying the licensing and (2) the licensee-usage patterns, which altogether determine the actual costs according to the cost model.

In order to determine licensing costs, several possibilities exist. One well-known cost model will be explained below. A company is considered, which is organized in different

entities that can be cost centers. The company has decided that it needs, let us say, twenty licenses for, for example, a word processing software residing on a network server. The company has, for example, forty potential users of the  
5 software product. Twenty licenses allow twenty users to concurrently have access to the software program on the server. When the twenty-first user wants to have an access to the program, the access is denied, since the company only purchased twenty licenses. When, on the other hand, none of  
10 the users or only a small number of users uses the software product concurrently, this does not result in a cost-reduction for the licenses. This is the case, since the cost model states that the company has to pay for the twenty licenses, irrespective of the number of users using the software  
15 product. In this case, the cost-distribution to different cost centers is generally made such that each cost center is assigned the same portion of the overall costs, or is assigned a portion of the overall costs according to the number of users belonging to each cost center.

20

This license model has two problems. One problem is that, if more than twenty licenses are required, it is not possible to obtain a twenty-first access to the program product. On the other hand, when there are only a few accesses to the product,  
25 the costs do not decrease, but are the same, irrespective of the actual usage of the twenty licenses.

A new licensing model has recently been developed. This licensing model depends on peak usage and not on the number of purchased licenses or, alternatively, on the amount of license 5 checkout times. In this license model, the customer is granted an option for a very high number of licenses. What the customer has to pay for is, however, the maximum number of simultaneous accesses, i.e., licenses, to the product in a certain billing interval. This cost model does not consider 10 any usage times, etc., but is only directed to the maximum number of simultaneous accesses, i.e., accesses in a certain time window during the time interval.

Summary of the Invention:

15 It is accordingly an object of the invention to provide a method and an apparatus for determining a portion of total costs of an entity which overcome the above-mentioned disadvantages of the prior art methods and devices of this general type, which achieve a fair cost distribution among 20 different entities having access to the same product.

With the foregoing and other objects in view there is provided, in accordance with a first embodiment of the invention, a method for determining a portion of total costs 25 for each entity of a plurality of entities. The total costs relate to a total maximum number of accesses of all the

entities occurring in a predetermined time window over a time interval having a plurality of time windows, to a product, and there exists a predetermined relationship between the total maximum number of accesses of all the entities and the total costs. The method includes obtaining, for each of the entities, a maximum value relating to a maximum number of accesses of the entity in all the time windows over the time interval resulting in a group of maximum values for the plurality of entities. The smallest maximum value is found among the group of maximum values for all the entities. Costs for the smallest maximum value are distributed among all the entities having access to the product in the time interval. A next-higher maximum value is found among the group of maximum values. A difference in costs for the smallest maximum value and the next-higher maximum value is determined from the predetermined relationship. The difference in costs is distributed among the entities having an equivalent or a higher maximum value than the next-higher maximum value. A further next-higher maximum value is repeatedly found from the group of maximum values and a determined difference in costs for the further next-higher maximum value is distributed among the entities having an equivalent or a higher maximum value. The costs obtained in the steps of distributing are accumulated for each of the entities individually to obtain an accumulated cost result. The portion of the total costs for

each of the entities based on the accumulated cost result for each of the entities are output.

As has been outlined above, the total costs for the product

5 are determined by the total maximum number of accesses in a predetermined time window. The predetermined time window can have different lengths dependent on the actual license model.

A length suitable for one case can be one second, while a length suitable for another case can be as long as one hour.

10 Accesses from different entities that take place during the predetermined time window are also termed "simultaneous" accesses. The simultaneous accesses to the product form an access peak or simply a peak, the peak height being equal to the number of simultaneous accesses, i. e. the accesses that  
15 take place in the predetermined time window.

The present invention is based on the finding that cost-transparency and fairness can only be obtained when those single-user access peaks that contribute to the overall peak

20 are determined and considered when distributing costs. It is, however, not fair to distribute all the costs to the entity or user group who caused the overall maximum peak, since then, the other entities that incidentally have a lower access peak would not have to pay anything. A fair cost distribution is,  
25 therefore, achieved when, based on the costs staircase, i.e., the predetermined relationship between the maximum number of

accesses and license costs, the costs for a peak or maximum value of one entity are e. g. equally distributed to all entities having the same or a higher maximum number of accesses in the time interval. The entity having a smaller  
5 maximum value than the peak, however, does not have to pay for the peak of the other entity. This feature clarifies the peak-oriented costs distribution that is performed step-by-step based on determining cost slices for each entity and accumulating the determined cost slices for each entity  
10 individually.

When the maximum peak on which the overall costs are based is not caused by a single entity alone, but is caused by accesses of multiple entities, this so-called multi-user peak is broken  
15 down into single-user or single-entity peaks. This is performed for all multi-user peaks in a given time interval.

When the maximum values for the different entities are extracted, the costs are distributed as described above. The so-called residual costs or allocation costs arising from the  
20 fact that more than one entity contributed to the peak decisive for billing are distributed equally among the entities.

In another embodiment of the present invention, hierarchies  
25 can be considered. The term hierarchy stands for a system in which an entity includes several sub-entities. The costs of

each entity are distributed based on the fixed relationship between the maximum number of accesses and costs. The sub-distribution among sub-entities is performed similar to the original distribution among entities. While the original  
5 distribution is directly based on the costs staircase, i.e., the predetermined relationship between accesses and costs, the sub-distribution is only indirectly related to the fixed relationship. In particular, only ratios of the fixed relationship are used, rather than absolute values. The cost  
10 sub-distribution is performed by using the cost portion distributed to an entity multiplied by certain ratios derived from the predetermined relationship between accesses and costs.

15 With the foregoing and other objects in view there is provided, in accordance with a second embodiment of the invention, a method for determining a portion of total costs for each entity of a plurality of entities. The total costs relate to a total maximum number of accesses for all of the  
20 entities occurring in a predetermined time window over a time interval having a plurality of time windows, to a product, and there exists a predetermined relationship between the total maximum number of accesses of all the entities and the total costs. The method includes obtaining, for each of the  
25 entities, a maximum value relating to a maximum number of accesses of the entity in all the time windows over the time

interval resulting in a group of maximum values. A highest maximum value is found from the group of maximum values for the entities and a next-lower maximum value is found from the group of maximum values. A difference in cost is determined  
5 for the highest maximum value and the next-lower maximum value from the predetermined relationship. The difference in cost is distributed to the entity having the highest maximum value. Another next-lower maximum value is repeatedly found among the group of maximum values and a determined difference for the  
10 next-lower maximum value is distributed among the entities having an equivalent or a higher maximum value. The costs obtained in the steps of distributing are accumulated for each of the individual entities to obtain an accumulated cost result and the portion of the total costs are output for each of  
15 the entities based on the accumulated result for each of the entities.

With the foregoing and other objects in view there is provided, in accordance with a third embodiment of the  
20 invention, an apparatus for determining a portion of total costs for each entity of a plurality of entities. The total costs relate to a total maximum number of accesses of all the entities occurring in a predetermined time window over a time interval having a plurality of time windows, to a product, and  
25 there exists a predetermined relationship between the total maximum number of accesses of all the entities and the total

costs. The apparatus contains a processing unit programmed to:

a) obtain, for each of the entities, a maximum value relating to a maximum number of accesses of the entity in all time windows over the time interval resulting in a group of maximum values;

b) find the smallest maximum value among the group of maximum values for the entities;

c) distribute costs for the smallest maximum value among all the entities having access to the product in the time interval;

d) find a next-higher maximum value among the group of maximum values;

e) determine a difference in costs for the smallest maximum value and the next-higher maximum value from the predetermined relationship;

f) distribute the difference in costs among the entities having an equivalent or a higher maximum value than the next-higher value;

10

15

20

25

g) repeatedly find a further next-higher maximum value among the entities and distribute a determined difference in costs for the further next-higher maximum value among the entities having an equivalent or a higher maximum value; and

h) accumulate costs obtained for each of the entities individually to obtain an accumulated cost result and output the portion of the total costs for each of the entities based on the accumulated result for each of the entities.

With the foregoing and other objects in view there is provided, in accordance with a fourth embodiment of the invention, an apparatus for determining a portion of total costs for each entity of a plurality of entities. The total costs relate to a total maximum number of accesses of all the entities occurring in a predetermined time window over a time interval having a plurality of time windows, to a product, and there exists a predetermined relationship between the total maximum number of accesses of all the entities and the total costs. The apparatus includes a processing unit programmed to:

a) obtain, for each of the entities, a maximum value relating to a maximum number of accesses of the entity in

all the time windows over the time interval resulting in  
a group of maximum values for the entities;

5           b) find the highest maximum value among the group of  
maximum values for the entities;

c) find a next-lower maximum value among the group of  
maximum values;

10          d) determine a difference in cost for the highest maximum  
value and the next-lower maximum value from the  
predetermined relationship;

15          e) distribute the difference costs to the entity having  
the highest maximum value;

20          f) repeatedly find another next-lower maximum value among  
the group of maximum values and distribute a determined  
difference for the another next-lower maximum value among  
the entities having an equivalent or a higher maximum  
value; and

25          g) accumulate the costs obtained for each of the entities  
individually to obtain an accumulated cost result and  
output the portion of the total costs for each of the

entities based on the accumulated result for each of the entities.

In accordance with an added mode of the invention, there is  
5 the step of calculating the total costs by combining the total maximum numbers of accesses in more than one time interval.

In which the maximum value for each of the entities obtained in the obtaining step is calculated by using a maximum of the maximum number of accesses in the more than one time interval  
10 as the maximum value for the entity for the more than one time interval.

In accordance with another mode of the invention, there is the step of performing the repeatedly finding step until a highest  
15 maximum value among the entities is processed in the case where the total maximum number of accesses of all the entities is equal to the maximum value of a single one of the entities.

In accordance with an additional mode of the invention, in a  
20 case where the total maximum number of accesses is produced by more than one of the entities, performing the following steps before performing the outputting step: calculating a residual cost being a difference between the total maximum number of accesses and a highest maximum value of a single one of the  
25 entities; and distributing the residual cost among all the entities equally, and the portion of the total costs for each

of the entities equals the accumulated cost result plus a part of the residual costs distributed to each of the entities.

In accordance with a further mode of the invention, there are

5 the steps of organizing at least one of the entities into a plurality of sub-entities; providing a maximum value of accesses for each of the sub-entities; and distributing the portion of the total costs of the entity among the sub-entities based on percentage ratios derived from the

10 predetermined relationship.

In accordance with a further added mode of the invention,

there are the steps of performing the distributing of the portion of the total costs of the entity step with the following sub-steps: calculating a ratio of the costs for the maximum value of a sub-entity to the costs for the maximum value of the entity from the predetermined relationship; calculating a cost amount by multiplying the ratio by the portion of the total costs for the entity; and distributing 20 the cost amount among the sub-entities having a maximum value equal to or greater than the maximum value, on which the cost amount is based.

In the invention, the costs are license costs for use of the

25 product and the entities are formed as cost centers.

Alternatively, the entities can be formed as business units, and the sub-entities are formed as the cost centers.

In accordance with a concomitant feature of the invention,

5 there are the steps of allowing a plurality of users to have access to the product; attributing each of the users to one of the entities; and performing the obtaining steps to include sub-steps. The sub-steps include counting a number of accesses of the users attributed to the entity in subsequent 10 time windows to obtain a series of access numbers for the time interval; and searching for a maximum access number in the series to obtain the maximum value for the entity in the time interval.

15 Other features which are considered as characteristic for the invention are set forth in the appended claims.

Although the invention is illustrated and described herein as embodied in a method and an apparatus for determining a 20 portion of total costs of an entity, it is nevertheless not intended to be limited to the details shown, since various modifications and structural changes may be made therein without departing from the spirit of the invention and within the scope and range of equivalents of the claims.

The construction and method of operation of the invention, however, together with additional objects and advantages thereof will be best understood from the following description of specific embodiments when read in connection with the 5 accompanying drawings.

Brief Description of the Drawings:

Fig. 1 is a block diagram illustrating a general setting in which the present invention can be carried out;

10 Fig. 2 is a block diagram illustrating the method in two embodiments according to the invention;

15 Fig. 3 is a bar graph illustrating an example of a predetermined relationship between a maximum number of accesses and total costs;

Fig. 4 is a graph illustrating an inventive distribution of costs to the entities;

20 Fig. 5 is a graph illustrating a number of multi-user peaks over a time interval;

25 Fig. 6 is a graph illustrating multi-group peaks from Fig. 5, after breaking down to the peaks of the single entities and after extracting the maximum values for each entity;

Fig. 7 is a chart for explaining the inventive method when the license model includes peak averaging;

5 Fig. 8a is a graph showing multi-group peaks caused by sub-entities in a hierarchical organization; and

Fig. 8b is a graph showing business unit peaks for the example shown in Fig. 8a.

10

Description of the Preferred Embodiments:

Referring now to the figures of the drawing in detail and first, particularly, to Fig. 1 thereof, there is shown an overview of the setting, in which the present invention may be carried out. Three entities exists, 11, 12, 13, which may or may not be organized into sub-entities. For illustration purposes, the first entity 11 is shown to contain three sub-entities, 11a, 11b and 11c. All entities or sub-entities have an access 15 to a product 17.

20

A license model 19 exists which calculates the total costs for using the product based on a maximum number of simultaneous accesses in a certain billing time interval. The license model is based on a cost model 20 that includes a so-called cost staircase that is a predetermined relationship between a maximum number of accesses and the total costs for the maximum

number of accesses. The detailed cost model 20 is shown in Fig. 3.

The license model 19 receives, as input, via line 21, a  
5 maximum number of simultaneous accesses of all the entities or  
the sub-entities within the time interval. As another input,  
the license model 19 receives, via line 22, a cost value for  
the received maximum number of simultaneous accesses and  
outputs, via line 23, the total costs for all the entities  
10 using the product 17 and for a certain billing interval.

PROTOTYPING &amp; DESIGN OF SYSTEMS

The problem now exists as to how to distribute the total costs  
output on line 23 by the license model 19 to the different  
entities. Of course, the costs could be equally distributed  
15 among the different entities, irrespective of the actual work  
of the entities. This cost distribution is regarded as  
unfair, since all entities have to pay the same amount,  
although, for example, the third entity 13 has not used the  
product at all during the billing interval. It is assumed  
20 that the first entity 11 was the only entity using the product  
during the billing interval. Additionally, it is assumed that  
the first entity 11 includes a certain amount of users to  
concurrently access to the product that resulted in a high  
peak and, therefore, high costs. When the users of the first  
25 entity 11 know that the total usage costs for the product are  
equally distributed among all the entities 11, 12, 13, they do

not have a special interest in changing their access habits, since the amount, the first entity 11 has to pay is always the same, and it does not matter whether the users try to serialize their needs for the product for reducing license 5 costs or whether the users of the first entity 11 do not think at all of a fair usage of resources.

In Fig. 1 there is shown a block 30 that includes the inventive apparatus and method for determining a portion of 10 the total costs for each of the entities 11, 12, 13 or sub-entities, if an entity includes sub-entities. Block 30 outputs, via lines 31, 32 and 33, the cost portions for the first, second and third entities 11, 12 and 13. The costs allocated for the first entity 11 can be, as will be explained 15 later, again sub-distributed to the sub-entities 11a, 11b and 11c.

Before referring to Fig. 2, reference is made to Fig. 3 illustrating the so-called costs staircase or fixed 20 relationship between the maximum number of accesses and costs. Along the horizontal axis in Fig. 3, the total maximum numbers of accesses of all the entities to the product from 1 to 8 are shown. Along the vertical axis, the license costs are given. When the total maximum number of accesses of all the entities 25 equals to 1, the total costs are one unit. However, when the total maximum number of accesses of all entities equals to 3,

the total costs are 5 units. Additionally, when the total maximum number of accesses of all the entities to the product is 4 or 5, the price is the same, i.e., 8 units. The costs staircase given in Fig. 3 illustrates a certain example for an 5 arbitrary license model based on access peaks. Generally, license models will have a kind of a monotonic structure, which results that, in general, the total costs increase with an increasing number of the total maximum number of accesses over a time interval. Naturally, this predetermined 10 relationship does not have to be a linear relationship, but as it is shown in Fig. 3, generally includes non-linearities.

The staircase in Fig. 3 is constructed, when each possible number between 1 and 8 in Fig. 3 is taken and compared to the 15 license model. Note that these are the hypothetical license costs for an overall peak of between 1 and 8 or, generally, for an overall peak n between 1 and p.

It becomes clear from Fig. 3 that for certain values of n, the 20 costs rise from one level to the next level. Additionally, all values of n that yield the same overall costs can be grouped. The range (1..p) is now sub-divided into a "list" of smaller ranges, for each of which the cost is the same for all its members (1..s<sub>1</sub>), (s<sub>1</sub>+1..s<sub>2</sub>), . . . , (s<sub>k-1</sub>+1..p). These 25 ranges correspond to the plateaus of the Fig. 3 staircase, while C<sub>i</sub> is the cost corresponding to a certain range (s<sub>i</sub>..

$s_1 \dots s_i$ ). Therefore, for  $C_1$  the first range equals 1, while, for example,  $C_7$  equals 13.

In the following, reference is made to Fig. 2 to illustrate  
5 the inventive method. In step 40, a maximum number of  
accesses of each of the entities 11, 12, 13 in the time window  
over the considered time interval is obtained. In block 42,  
the lowest or highest maximum value among the entities is  
found. First, the left branch of Fig. 2 is considered. In  
10 this case, step 42 is performed to find the lowest maximum  
value among the entities.

For a better explanation of the inventive method, reference is  
made to Fig. 4. Along the vertical axis in Fig. 4, the  
15 maximum number of accesses in a time interval for 4 entities  
or user groups UG1, UG2, UG3 and UG4 is illustrated. The  
lowest maximum number of accesses is from user group number  
UG4. User group number UG4 has a maximum of 4 accesses over  
the interval. User group UG3 has a maximum value of 6  
20 accesses, user group UG1 has a maximum value of 7 accesses and  
user group UG2 has a maximum number of 8 accesses.

In step 42 of Fig. 2, user group UG4 will be identified in the  
present example. The maximum value of accesses of user group  
25 UG4 is 4. Four overall peaks correspond to 8 cost units in  
accordance with the Fig. 3 staircase. Step 44 of Fig. 2 is

operative to distribute the total costs for the lowest maximum value among entities having an equal or higher maximum value.

In the Fig. 4 example, all the entities, i.e., user groups, have the same or a higher number of maximum values.

5 Therefore, the costs, i.e., 8 units, for a number of 4 accesses to the product are equally distributed among all user groups, as it can be seen from Fig. 4. Thus, the corresponding cost slice allocated to user groups UG1, UG2, UG3 and UG4 is created.

10

A next higher maximum value has then to be identified. In the Fig. 4 example, this will result in an identification of user group UG3 having 6 overall peaks in the time interval. Six overall peaks result in 12 cost units. Since 8 cost units have already been distributed, only 4 additional units have to be distributed among user group UG1, user group UG2 and user group UG3, but not for user group UG4, since user group UG4 has a maximum value lower than the maximum value of user group UG3.

15  
20

Referring to Fig. 2, the next step following step 44 is to determine the cost difference for the next higher value (step 46). In the Fig. 4 embodiment, this cost difference equals 4 cost units. Therefore, step 46 will output 4 cost units. In step 48, these 4 cost units are distributed among user group UG1, user group UG2 and user group UG3, such that each user

group receives 4/3 cost units. In this way, the corresponding cost slice allocated to user group UG1, user group UG2 and user group UG3 is generated. This proceeding, i.e., finding the next higher value, determining the cost difference for the 5 next higher value and distributing this cost difference among the respective entities is repeated until all maximum values of the entities are processed.

Turning back to Fig. 4, the next higher maximum value is 7, 10 i.e., user group UG1 having an overall peak number of 7, will be identified. Seven accesses correspond to 13 cost units, which is 1 cost unit more than the 12 units corresponding to 6 accesses and user group UG3. Therefore, the cost difference for the next higher value equals to one unit. This one unit 15 is distributed among user group UG1 and user group UG2, since user group UG3 and user group UG4 have lower maximum values than 7 accesses. Therefore, the user group UG1 and user group UG3 each receive 0.5 cost units such that another 20 corresponding cost slice for user group UG1 and user group UG3 is completed.

Finally, the next higher value is found, which will identify user group UG2 as having 8 accesses to the product. 8 accesses correspond to 17 cost units, which are 4 cost units 25 more than the number of cost units for 7 accesses. Since user group UG2 has the highest maximum value, the 4 units, i.e.,

the cost difference for 8 accesses with respect to 7 accesses,  
is distributed to user group UG2 only.

In a final step of accumulating 50, the costs distributed to  
5 the different user groups are accumulated such that portions  
for the different entities or user groups are output as  
indicated in the left corner of Fig. 4.

It can be seen from the cost distribution from Fig. 4 that  
10 user group UG2 has to pay the largest portion of total costs,  
this portion amounting to 7.833 units, since this user group  
caused a peak, while user group UG4 only has to pay 2 cost  
units, since this user group only had 4 accesses to the  
product in the billing time interval.

15 It also becomes clear from the cost distribution in Fig. 4  
that this cost distribution is much more fair when compared to  
the case in which user group UG2 would have to pay all costs,  
i.e., 17 units. The cost distribution is also much more fair  
20 when compared to the case in which the 17 units are equally  
distributed among the 4 user groups which would result in 4.25  
cost units for each user group. The cost distribution would  
be unfair regarding user group UG4 and would be "too cheap"  
for user group UG2.

In other words, the inventive cost distribution scheme works as follows. For allocating the cost in a fair way, we start with the first range ( $1..s_1$ ). We take each user group with a maximum peak equal or greater than  $s_1$ ; assume that there are  $g_1$  of those groups altogether. In fact, these are all user groups that have used product p at all during the given month. Divide the costs  $C_1$  by  $g_1$  and allocate  $C_1/g_1$  for each group involved.

As the next step, for the range ( $s_1+1..s_2$ ), we take each user group with a maximum peak equal or greater than  $s_2$ ; assume there are  $g_2$  of these groups. These are all groups that cause the overall peak to be equal or greater than  $s_2$ . Divide the costs  $C_2 - C_1$  by  $g_2$  and allocate  $(C_2 - C_1)/g_2$  for each group involved. Note that we allocate  $C_2 - C_1$  and not  $C_2$  because we have allocated  $C_1$  already.

In general, for each range ( $s_{i-1}+1..s_i$ ), ( $1 \leq i \leq k$ , with  $s_0 = 1$  and  $s_k = p$ ), we determine the  $g_i$  groups having a maximum peak equal or greater than  $s_i$  and allocate  $(C_i - C_{i-1})/g_i$  to each of these groups, ( $C_0 = 0$  and  $C_k = C$ ).

As it can easily be seen, corresponding to the ranges, the overall cost  $C$  is partitioned into slices  $C_1, C_2 - C_1, C_3 - C_2, \dots, C_k - C_{k-1}$ , the sum of which equals to the overall cost  $C$ . Furthermore, for each cost slice, at least one user group

participates in the cost allocation for that slice; thus, all cost slices are allocated. From our precondition of Step 1, at least one user group has a maximum peak equal to the overall peak  $p$ ; consequently for each range  $(s_{i-1}+1..s_i)$ , the 5 peak  $p$  of the user group is equal or greater than the upper limit  $s_i$ . Thus, this group will participate in the cost allocation of the cost  $C_i$  for that range.

The overall result of this stratified cost-allocation scheme 10 is that the entire costs are allocated - the scheme is correct - and that each user group pays at most only those costs that it would be responsible for, had it been alone in using the product.

15 In the basic scheme described with respect to Fig. 3 and Fig. 4, it was assumed that there are only peaks that are created by a single user group. In particular, it presupposes that the overall peak  $p$  is reached by at least one of the user groups. In accordance with another embodiment of the present 20 invention, this assumption is given up and so called multi-group peaks are considered. It is assumed that two user groups checking out licenses for a product  $p$  at the same time. One group takes 4 licenses, the other group takes 6 licenses. This results in an overall peak of 10 licenses. If the basic 25 scheme as outlined with respect to Fig. 4 would be used without modification, the method would be finished at the cost

for the group 2 peak of 6, since this is the highest individual peak. The difference between its cost and the costs for the overall peak of 10 would remain unallocated. In accordance with the present invention, the difference between 5 the highest individual peak and the overall peak is equally distributed among all user groups. The logic behind this feature is that the whole community of user groups or entities should be responsible for enabling coincidental shared product usage. The costs that are equally distributed among all user 10 groups are also termed common allocation costs.

10  
15

S&Z-10011001

for the group 2 peak of 6, since this is the highest individual peak. The difference between its cost and the costs for the overall peak of 10 would remain unallocated. In accordance with the present invention, the difference between 5 the highest individual peak and the overall peak is equally distributed among all user groups. The logic behind this feature is that the whole community of user groups or entities should be responsible for enabling coincidental shared product usage. The costs that are equally distributed among all user 10 groups are also termed common allocation costs.

For a detailed explanation of multi-group peaks, reference is made to Fig. 5. The horizontal axis of Fig. 5 shows the time axis, while the vertical axis of Fig. 5 shows the number of accesses in a time window along the time axis. It is assumed 15 that all multi-group peaks shown in Fig. 5 fall into the time interval. To handle the multi-group peaks, each multi-group peak is broken-down into single group peaks that contribute to a multi-group peak. The same procedure as described with 20 respect to Fig. 4 is then carried through. Thus, multi-group peaks are treated as many single group peaks and the maximum value of accesses of each single group or entity is determined in the time interval and used for cost distribution.

25 Fig. 6 shows this situation for the multi-group peaks from  
Fig. 5. It becomes clear from Fig. 6 that user group UG1 only

had a maximum of 1 access in the time interval. User group UG2 had a maximum of 2 accesses and user groups 3 and 4 had a maximum of 3 accesses. With respect to the overall costs, it becomes clear from Fig. 5 that the overall costs are the costs  
5 to be paid for 7 accesses.

The cost distribution works as follows. First, the costs for 1 to 3 simultaneous accesses are distributed as outlined with respect to Fig. 4. Then, the costs for 7 accesses less the  
10 costs for 3 accesses remain unallocated. These costs are treated as common allocation costs. They are distributed among all user groups in equal portions. The difference of the costs for the highest single-group peak and the costs for the overall peak, which is a multi-group peak in the above-  
15 identified example, is equally distributed among all user groups. The argument behind this is that the whole community of entities should be responsible for enabling coincidental shared product usage. As has been outlined above, this equally-distributed part of the license costs are termed  
20 common allocation costs.

The embodiments discussed so far allocate the costs of the overall peak from a single time interval. However, license contracts also exist, that usually consider several  
25 consecutive time intervals and take the average of the corresponding overall peaks to determine license costs. This

procedure levels out unusual or coincidental peaks. If overall peaks are averaged, it seems to be natural to average user-group peaks, too. After averaging, one could apply the approach described in connection with Fig. 4. Although this 5 seems to be the most natural approach, it is not adequate. In order to clarify this, reference is made to Fig. 7. It is assumed that two user groups, i.e., UG1 and UG2 work with product P during two months, April and May. User group UG1 has a peak of 8 in April and does not use P in May. User 10 group UG2 has no use for P in April, but has peak usage of 6 for P in May. Now, the average of the overall peaks yields 7, whereas the averages of the group peaks yield 4 and 3, respectively. As it can be seen from Fig. 7, the difference in costs between the highest average user peak of 4 and the 15 average overall peak of 7 remains unallocated and would be equally distributed among all user groups, although user group UG1 and user group UG2 clearly caused the corresponding overall peaks of that average, while other groups apart from user group UG1 or user group UG2 did not use any licenses at 20 all. Instead of being cause-oriented, this procedure would introduce arbitrariness.

In accordance with the present invention, in order to increase justness and transparency in cost allocation, another approach 25 is applied. For all time intervals, over which the average will be calculated, each user group is assigned the maximum of

its interval peaks. This results in that - conceptually - these intervals are united and the maximum peak for each user group for the resulting united interval is selected as a maximum value for the entity. The algorithm described in  
5 connection with Fig. 4 is then used for cost distribution.

Each cost slice is allocated to the participating user groups. In contrast to the Fig. 4 embodiment, however, the cost allocation will stop at the overall peak, which is the calculated average for the two time intervals April and May.  
10

When the average is not an integer, the contracts for products P will state how to calculate the corresponding last cost slice. When averaging over more than one time interval, there can also be the case that the maximum user group peak is higher than the (averaged) overall peak. In this case, the  
15 minimum out of the overall peak and the maximum user group peak has to be formed and used as the maximum value for the respective user group. In this way, the maximum value used for cost stratification is related to the maximum number of accesses, wherein the relation consists in the mathematical  
20 operation of minimum formation.

It is to be noted that all single-group peaks that are higher than this average will be treated as if they were equal to this average. In the above example illustrated in Fig. 7, the  
25 average of the overall peak equals 7. For cost distribution, user group UG1 will participate in all slice allocations until

overall peak 7 is processed. After that, the algorithm will stop. Thus, even though user group UG1 has a peak of 8, it is treated as if the peak were 7. This embodiment is advantageous in that costs resulting from high peaks are still

5 allocated among those who have caused the peaks. Furthermore, the averaging reduces the maximum peak value considered.

Thus, the goals of cause-orientation and comprehensiveness, while incorporating the leveling effect of averaging for the license costs is still maintained. Another advantage is that

10 the Fig. 7 method is also applicable to multi-user groups. By this method, there does not occur a leveling-out of user-group peaks, but the leveling-out of the overall peak is "propagated" to the single user peaks.

15 In the following, another embodiment of the present invention is described which constitutes a recursive generalization of the inventive cost distribution method for hierarchical organizations. As has been outlined with respect to Fig. 1,

20 an entity can be divided into several sub-entities, resulting in a hierarchic structure. While the embodiments described thus far consider each access dimension, like site and user

group to be flat with respect to peak summation, the following embodiment is able to work with hierarchically-organized dimensions, in which high-level units do not play an

25 autonomous role within license-cost allocation. The inventive cost distribution for hierarchically-structured dimensions

results in a distribution of the common-allocation costs adequate to hierarchies. The basic cost-allocation method for each hierarchy level is the same as described in connection with Fig. 4.

5

If license check-out dimensions, like products, site and user group, are flat, cost distribution as described above yields an optimum approach with respect to cause orientation and cost transparency. When such a dimension is actually

10 hierarchically organized, the subsequently explained embodiment allows an even more cause-orientated and transparent way. By way of example, the hierachic user groups consist of cost centers (CC) and business units (BU). Additionally, it is assumed that a multi-group peak results  
15 from two cost-center peaks of 6 and 4, respectively that belong to one business unit. In the basic embodiment, these cost-center peaks do not sum up to a peak of 10 for the business unit, although their check-outs happen simultaneously. Instead, the business unit peak is equal to  
20 6, the maximum of both cost-center peaks. The remaining cost difference between the maximum cost-center peak of 6 and the overall peak of 10, i.e., the common-allocation costs, would be evenly distributed among all cost units participating in the entire license-cost allocation, not just within that  
25 multi-group peak. The reason for this way of calculation is that cost-stratification as described with respect to Fig. 4

originally works with flat dimensions only and does not take hierarchies into account. The above example clearly shows that the multi-group peak of 10 is solely caused by the business unit, to which these two cost centers belong. Thus,  
5 it would be much more fair to allocate the cost of that peak completely to this business unit instead of evenly distributing it among all cost centers taking part in the entire process. As a consequence, there would be no common-allocation costs at this stage of allocation. In general, it  
10 is the goal of the present invention to reduce the common-allocation costs as much as possible and to apply the cost distribution based on cost slices (see Fig. 4).

In general, in hierarchical structures, the license costs are  
15 distributed among entities of the highest level. The costs for each entity are then distributed among the sub-entities of this entity. This hierarchical recursive algorithm can be applied for as many hierarchical levels as required, in that the costs for a sub-entity can be distributed among a  
20 plurality of lower-level sub-entities depending from the sub-entity.

Continuing the above example, the costs now allocated to a business unit have to be distributed among the cost centers of  
25 the business unit. For simplicity, it is assumed that the two cost centers mentioned above are the only cost centers of that

business unit. Cost centers are now the smallest elements of the user-group hierarchy considered and, furthermore, cost centers are the only kind of user groups for sub-entities, the allocation has to deal with at this level of the process.

5 Therefore, all preconditions necessary for the optimal allocation of the original cost-stratification method are now present. Costs are stratified until the maximum cost-center peak of 6 and the remaining difference of 4 is evenly distributed among both cost centers. The following two points  
10 are especially remarkable:

a) Hierarchies are important with respect to common-allocation costs. They allow one to treat higher-level peaks as the sum and not the maximum, of the corresponding lower-level peaks.

15 As a consequence, common-allocation costs are propagated as far as possible down the entity hierarchy, thereby minimizing their amount. This clearly improves transparency, and cause-orientation.

20 b) The generalized cost-allocation algorithm is recursively applied along the user-group hierarchy until its lowest level is reached. Thus, the preferred embodiment is a natural recursive generalization of the basic cost-stratification method. One aspect that has to be mentioned in this regard is  
25 that the absolute values of the cost staircase, i.e., the costs for a certain number of accesses, can only be applied

for cost stratification in the highest level of hierarchy. As will be outlined in detail below, for the second and the following levels, the fixed relationship has to be applied using the percentage relations between cost steps, rather than  
5 the absolute values.

In order to illustrate the hierarchical cost distribution embodiment of the present invention, reference is made to Figs. 8a and 8b. It is assumed that a business unit BU1 includes cost centers CC1 and CC2 and that another business unit BU2 includes cost centers CC3 and CC4. Additionally, the scenario given with respect to Fig. 8a is considered. Two multi-group peaks 80 and 82 exist in a certain time interval. The multi-group peak 80 has a peak value of 15, while the multi-group peak 82 has a peak value of 12. Thus, the overall costs are determined by the multi-group peak 80. Naturally, one could distribute the costs for the peak value of 15 among all 4 cost centers as described with respect to Fig. 4.

However, this would not take the hierarchical structures into account. In particular, one wants to know which are the costs for each business unit. Therefore the business unit hierarchy level takes responsibility for the "peaks" caused by all its associated cost centers.

25 In order to distribute the costs among the business units 1 and 2, the multi-group peaks 80 and 82 are broken-down into

"business unit peaks". Thus, there results business unit peaks 84 and 86 from the multi-group peak 80, and business unit peaks 88 and 90 from the multi-group peak 82. Since the costs for the business units have to be determined, the

5 maximum business unit peak for business unit BU1 and the maximum business unit peak for the business unit BU2 have to be determined from the chart shown in Fig. 8b. The maximum peak for business unit BU1 is peak 84 having a peak value of 8, while the maximum peak for business unit BU2 is peak 90,  
10 having also a peak value of 8. In this case, the cost distribution is very simple. Each business unit has to pay half the price, i.e., half the amount given by a cost staircase for a peak value of 15.

15 Now, the first recursion or the cost allocation to the entities in the highest hierarchy level is completed. The next step will be to allocate the costs of each business unit among the cost centers. Therefore the costs for the business unit BU1 has to be distributed among cost centers CC1 and CC2  
20 and the costs for business unit BU2 have to be distributed among cost centers CC3 and CC4. At this level, business unit peaks are multi-group peaks, since they consist of the peaks of their associated cost centers. The "multi-group peaks" from Fig. 8b now have to be broken-down into single group  
25 peaks to identify the maximum peaks of each group. When the peaks in Fig. 8b are then broken-down, it becomes clear that

CC1 has a maximum peak of 5, CC2 has a maximum peak of 4, CC3 has a maximum peak of 7 and CC4 has a maximum peak of 5. It is to be noted that the maximum peak of CC2 is not contained in the maximum peak of business unit BU1. In general, a 5 maximum peak of a higher-level group need not contain even a single maximum peak of any of its lower-level members. This is a natural consequence of the fact that the overall peak need not contain any maximum single-group peaks.

10 The maximum peaks named above are now used to distribute the business-unit cost. Thus, for this application level of the stratification method, the costs for each business unit take the place of the top-level overall costs. In order to derive their share for each cost center, the original cost staircase  
15 is used, but now with the percentage relations between the steps of the staircase, rather than the absolute values of the staircase (the absolute values were used in the first recursion step). Now, the contract pattern of cost increase per peak has to be transferred to the new situation. To this end, the contract costs of the highest maximum peak for each business unit is taken as 100% and the contract costs of each lower peak are expressed as percentages in relation to the costs of the highest maximum peak. In the example, the highest maximum peak for the business unit BU1 is 5. The cost  
20 fraction for the peak of CC2 of 3 is contract-cost(3)/contract-cost(5). By using these percentages, a key  
25

exists for distributing the costs of a business-unit among its cost centers.

The distribution of the business-unit costs among their cost centers will now be discussed in detail. The business unit BU1 has costs corresponding to a peak of 8. Of these costs, the costs for the maximum cost center peak of 5 are stratified among CC1 and CC2 in accordance with the Fig. 4 algorithm. From the remaining costs corresponding to the difference between the overall peak of 8 and the maximum cost center peak of 5, each cost center takes half the difference. The same is done for business unit BU2. The maximum cost-center peak here is 7. Thus, the costs corresponding to a peak of 7 are stratified among CC3 and CC4. Of the rest, the difference between the peak of 8 and the peak of 7, CC3 and CC4 each take one half. These difference costs are the common allocation costs.

In the following, the above example will be numerically set out based on the Fig. 3 fixed relationship between license costs and the number of accesses or "overall peaks". As has been outlined above, each business unit is responsible for half the costs of a multi-group peak of 15, which is not included in the Fig. 3 staircase. However, it is assumed that the price for a peak of 15 is US\$ 20,000. Thus, each business unit, i.e., business unit BU1 and business unit BU2 each have

to pay an amount of US\$ 10,000 for a maximum value of 8 licenses or accesses, each business unit had. Referring to the Fig. 3 cost staircase, the percentage for 5 licenses with respect to 8 licenses is 8 (the license costs for a peak having 5 accesses) to 17 (the license costs for a peak having 8 accesses). The fraction equals to 47% (8/17). The fraction for 3 licenses is 5 (the license costs for 3 licenses) to 17 (the license costs for 8 licenses). This fraction equals to a percentage of 29% (5/17). Thus, the percentage costs of 5 licenses are 47% of US\$ 10,000, i.e., US\$ 4,700. The percentage costs for 3 licenses are 29% of US\$ 10,000, i.e., US\$ 2,900. The basic cost allocation algorithm described with respect to Fig. 4 can now be applied. First, the costs for 3 licenses (cost center CC2) for business unit BU1 (peak 84 in Fig. 8b), i.e., US\$ 2,900 are distributed between the cost center CC1 and the cost center CC2. Thus, each cost center has a first cost slice of US\$ 1,450. The difference between the costs for 5 licenses (US\$ 4,700) and the costs for 3 licenses (US\$ 2,900), which amounts to US\$ 1,800, is attributed to cost center CC1 alone. The remaining difference, i.e., US\$ 10,000 - US\$ 4,700 = US\$ 5,300, are the common allocation costs, which are equally distributed among cost center CC1 and cost center CC2, such that each cost center receives an additional US\$ 2,650. Thus, the accumulated costs for cost center CC1 are US\$ 5,900 and the accumulated costs for cost center CC2 are US\$ 4,100. The sum

of the accumulated costs equals to US\$ 10,000 for business unit BU1.

In the above example, the recursion is complete, since only

5 two hierarchy levels exists, because the cost centers are considered atomic. They cannot be broken down into smaller units. As it can be seen, the overall costs for a peak of 15 are entirely allocated among business units and then further among cost centers.

10

The above algorithm can be generalised as follows:

Initialization: Calculate overall costs

parent level = top level of the hierarchical dimension

child level = next-lower level

Generalized Cost-Stratification Procedure:

20 Apply the basic cost-stratification algorithm treating each child-level group as single-group.

Assign allocated costs to the corresponding child-level groups if child level is not lowest level.

25

For each child-level group:

Take the highest maximum peak  $m$  of the group and calculate the new peak-to-cost relation by mapping each maximum peak  $i$  of the next-lower level contained in the group to (contract-cost-  
5 for-peak( $i$ )/contract-cost-for-peak( $m$ ) \* cost-allocated-to-the-group.

Apply the Generalized Cost-Stratification Procedure with new parent level = current child level and new child level = next-  
10 lower level.

It has to be outlined that not only user groups can be perceived as hierarchies, but also sites. The inventive generalized cost-stratification algorithm makes it easy to  
15 select one from a set of given hierarchical dimensions and to allocate overall costs according to this hierarchy.

Although the inventive cost distribution algorithm has been described in connection with Fig. 4 such that it is started  
20 from the lowest maximum value and then step-by-step propagated to the highest maximum number of an entity, this scheme can also be reversed. To this end, reference is made to the right branch in Fig. 2. When the cost distribution method is performed starting from the highest maximum value among the  
25 entities, the next lower maximum value has first to be found.

Regarding Fig. 4, this results in that, first, the user group UG2 having a peak value of 8 is identified and then the user group UG1, which has the next lower peak value. Then, the cost difference between the highest maximum value and the next 5 lower maximum value has to be determined (step 52 of Fig. 2).

The cost difference just determined is then distributed to user group UG2 alone, or generally, to the user groups having the same or a higher maximum value. Since user group UG2 has the highest maximum value, it exclusively receives the

10 difference that amounts in the Fig. 4 example to 4 units (step 54 in Fig. 2). The algorithm then proceeds with step 56 of finding a next lower maximum value. In this case, the user group UG3 from Fig. 4 would be identified. Then, again, a cost difference for this next lower maximum value is 15 determined and distributed (steps 52 and 54). This is indicated by a back-directed arrow 58 in the right branch of Fig. 2, which roughly corresponds to the recursive arrow 60 in the left branch of Fig. 2.

20 It becomes clear from the above that the inventive method of cost distribution can be carried through from the bottom of Fig. 4 to the top of Fig. 4 (user group UG4 to user group UG2) or, in reverse direction, from the top of Fig. 4 (user group UG2) to the bottom of Fig. 4 (user group UG4).